2.6 LATITUDE/LONGITUDE CODING USINGCOMPACT POSITION REPORTING (CPR)

2.6.1 PRINCIPLE OF THE CPR ALGORITHM

The Mode S extended squitters use compact position reporting (CPR) to encode latitude and longitude efficiently into messages.

Notes.—

1. The resulting messages are compact in the sense that several higher-order bits, which are normally constant for long periods of time, are not transmitted in every message. For example, in a direct binary representation of latitude, one bit would designate whether the aircraft is in the northern or southern hemisphere. This bit would remain constant for a long time, possibly the entire life of the aircraft. To repeatedly transmit this bit in every position message would be inefficient.

2. Because the higher-order bits are not transmitted, it follows that multiple locations on the earth will produce the same encoded position. If only a single position message were received, the decoding would involve ambiguity as to which of the multiple solutions is the correct location of the aircraft. The CPR technique includes a provision to enable a receiving system to unambiguously determine the location of the aircraft. This is done by encoding in two ways that differ slightly. The two formats, called even-format and odd-format, are each transmitted 50 per cent of the time. Upon reception of both types within a short period (approximately 10 seconds), the receiving system can unambiguously determine the location of the aircraft.

3. Once this process has been carried out, the higher-order bits are known at the receiving station, so subsequent single message receptions serve to unambiguously indicate the location of the aircraft as it moves.

4. In certain special cases, a single reception can be decoded into the correct location without an even/odd pair. This decoding is based on the fact that the multiple locations are spaced by at least 360 NM. In addition to the correct locations, the other locations are separated by integer multiples of 360 NM to the north and south and also integer multiples of 360 NM to the east and west. In a special case in which it is known that reception is impossible beyond a range of 180 NM, the nearest solution is the correct location of the aircraft.

5. The parameter values in the preceding paragraph (360 and 180 NM) apply to the airborne CPR encoding. For aircraft on the surface, the CPR parameters are smaller by a factor of 4. This encoding yields better resolution but reduces the spacing of the multiple solutions.

2.6.2 CPR ALGORITHM PARAMETERS AND INTERNAL FUNCTIONS

The CPR algorithm shall utilize the following parameters whose values are set as follows for the Mode S extended squitter application:

a) The number of bits used to encode a position coordinate, *Nb*, is set as follows:

For airborne encoding: *Nb* = 17

For surface encoding: *Nb* = 19

For TCP, TCP+1 encoding: *Nb* = 14.

Note 1.— The Nb *parameter determines the encoded position precision (approximately 5 m for the airborne encoding, 1.25 m for the surface encoding, and 41 m for the TCP, TCP+1 encoding).*

b) The number of geographic latitude zones between the equator and a pole, denoted *NZ*, is set to 15.

Note 2.— The NZ *parameter determines the unambiguous airborne range for decoding (360 NM). The surface latitude/longitude encoding omits the high-order 2 bits of the 19-bit CPR encoding, so the effective unambiguous range for surface position reports is 90 NM.*

The CPR algorithm shall define internal functions to be used in the encoding and decoding processes.

c) The notation **floor**(x) denotes the floor of x, which is defined as the greatest integer value k such that $k \leq x$.

*Note 3.— For example, **floor**(3.8) = 3, while **floor**(–3.8) = –4.*

d) The notation $|x|$ denotes the absolute value of x, which is defined as the value x when $x \geq 0$ and the value $-x$ when $x < 0$.

e) The notation **MOD**(x,y) denotes the "modulus" function, which is defined to return the value

$\text{MOD}(x,y) = x - y \cdot \text{floor}\left(\frac{x}{y}\right)$ where $y \neq 0$.

Note 4.— The value y *is always positive in the following CPR algorithms. When* x *is non-negative, **MOD**(x,y) is equivalent to the remainder of* x *divided by* y. *When* x *represents a negative angle, an alternative way to calculate **MOD**(x,y) is to return the remainder of* (x+360°) *divided by* y.

For example, MOD(–40°,6°) = MOD(320°,6°) = 2°.

f) The notation **NL**(x) denotes the "number of longitude zones" function of the latitude angle x. The value returned by **NL**(x) is constrained to the range from 1 to 59. **NL**(x) is defined for most latitudes by the equation,

$$\text{NL}(lat) = \text{floor}\left(2\pi \cdot \left[\text{arc}\ \cos\left(\frac{1 - \cos\left(\frac{\pi}{2 \cdot NZ}\right)}{\cos^2\left(\frac{\pi}{180°} \cdot |lat|\right)}\right)\right]^{-1}\right),$$

where *lat* denotes the latitude argument in degrees. For latitudes at or near the N or S pole, where the above formula would either be undefined or yield **NL**(*lat*) = 0, the value returned by the NL( ) function shall be 1. Likewise, at the equator, where the above formula might otherwise yield NL(*lat*) = 60, the value returned by the NL( ) function shall be 59.

Note 5.— This equation for NL( ) is impractical for real time implementation. A table of transition latitudes can be pre-computed using the following equation:

$$lat = \frac{180°}{\pi} \cdot \text{arc}\ \cos\left(\sqrt{\frac{1 - \cos\left(\frac{\pi}{2 \cdot NZ}\right)}{1 - \cos\left(\frac{2\pi}{NL}\right)}}\right) \text{for } NL = 2 \text{ to } 4 \cdot NZ - 1,$$

and a table search procedure used to obtain the return value for NL( ). The table value for NL = 1 is 90 degrees.

2.6.3 CPR ENCODING PROCESS

The CPR encoding process shall calculate the encoded position values XZ_i and YZ_i for either airborne, surface or TCP, TCP+1 latitude and longitude fields from the global position *lat* (latitude in degrees), *lon* (longitude in degrees), and the CPR encoding type i (0 for even format and 1 for odd format), by performing the following sequence of computations. The CPR encoding for TCP, TCP+1 shall always use the even format (i = 0), whereas the airborne and surface encoding shall use both even (i = 0) and odd (i = 1) formats.

a) Δlat_i (the latitude zone size in the N-S direction) is computed from the equation:

$$\Delta lat_i = \frac{360°}{4 \cdot NZ - i}$$

YZ_i (the <u>Y</u>-coordinate within the <u>Z</u>one) is then computed from Δlat_i and *lat* using separate equations:

For airborne encoding: $YZ_i = \text{floor}\left(2^{17} \cdot \frac{\text{MOD}(lat,\Delta lat_i)}{\Delta lat_i} + \frac{1}{2}\right)$

For surface encoding: $YZ_i = \text{floor}\left(2^{19} \cdot \dfrac{\text{MOD}(lat,\Delta lat_i)}{\Delta lat_i} + \dfrac{1}{2}\right)$

For TCP, TCP+1 encoding: $YZ_0 = \text{floor}\left(2^{14} \cdot \dfrac{\text{MOD}(lat,\Delta lat_0)}{\Delta lat_i} + \dfrac{1}{2}\right)$

b) $Rlat_i$ (the latitude that a receiving ADS-B system will extract from the transmitted message) is then computed from lat, YZ_i, and Δlat_i using separate equations:

For airborne encoding: $Rlat_i = \Delta lat_i \cdot \left(\dfrac{YZ_i}{2^{17}} + \text{floor}\left(\dfrac{lat}{\Delta lat_i}\right)\right)$

For surface encoding: $Rlat_i = \Delta lat_i \cdot \left(\dfrac{YZ_i}{2^{19}} + \text{floor}\left(\dfrac{lat}{\Delta lat_i}\right)\right)$

For TCP, TCP+1 encoding: $Rlat_0 = \Delta lat_0 \cdot \left(\dfrac{YZ_i}{2^{14}} + \text{floor}\left(\dfrac{lat}{\Delta lat_0}\right)\right)$

c) Δlon_i (the longitude zone size in the E-W direction) is then computed from $Rlat_i$ using the equation:

$$\Delta lon_i = \begin{cases} \dfrac{360°}{\text{NL}(Rlat_i) - i} & \text{when } \text{NL}(Rlat_i) - i > 0 \\ 360°, & \text{when } \text{NL}(Rlat_i) - i = 0 \end{cases}$$

d) XZ_i (the X-coordinate within the Zone) is then computed from lon and Δlon_i using separate equations:

For airborne encoding: $XZ_i = \text{floor}\left(2^{17} \cdot \dfrac{\text{MOD}(lon,\Delta lon_i)}{\Delta lon_i} + \dfrac{1}{2}\right)$

For surface encoding: $XZ_i = \text{floor}\left(2^{19} \cdot \dfrac{\text{MOD}(lon,\Delta lon_i)}{\Delta lon_i} + \dfrac{1}{2}\right)$

For TCP, TCP+1 encoding: $XZ_0 = \text{floor}\left(2^{14} \cdot \dfrac{\text{MOD}(lon,\Delta lon_0)}{\Delta lon_0} + \dfrac{1}{2}\right)$

e) Finally, limit the values of XZ_i and YZ_i to fit in the 17-bit or 14-bit field allotted to each coordinate:

For airborne encoding: $YZ_i = \text{MOD}(YZ_i, 2^{17})$,
$XZ_i = \text{MOD}(XZ_i, 2^{17})$

For surface encoding: $YZ_i = \text{MOD}(YZ_i, 2^{17})$,
$XZ_i = \text{MOD}(XZ_i, 2^{17})$

For TCP, TCP+1 encoding: $YZ_0 = \text{MOD}(YZ_0, 2^{14})$,
$XZ_0 = \text{MOD}(XZ_0, 2^{14})$

2.6.4 Locally unambiguous CPR decoding

The CPR algorithm shall decode a geographic position (latitude, $Rlat_i$, and longitude, $Rlon_i$,) that is locally unambiguous with respect to a reference point (lat_s, lon_s) known to be within 180 NM of the true airborne position (or within 45 NM for a surface message).

Note.— This reference point may be a previously tracked position that has been confirmed by global decoding (see 2.6.7) or it may be the own-aircraft position, which would be used for decoding a new tentative position report.

The encoded position coordinates XZ_i and YZ_i and the CPR encoding type i (0 for the even encoding and 1 for the odd encoding) contained in a Mode S extended squitter message shall be decoded by performing the sequence of computations given in 2.6.5 for the airborne and TCP, TCP+1 format types and in 2.6.6 for the surface format type.

2.6.5 Computations for the airborne message and TCP, TCP+1 message

The following computations shall be performed to obtain the decoded latitude/longitude for the airborne and TCP, TCP+1 message formats. For the TCP, TCP+1 format, i shall always be set to 0 (even encoding), whereas the airborne format shall use both even (i = 0) and odd (i = 1) encoding. For the airborne format, Nb shall equal 17, and for the TCP, TCP+1 format, Nb shall equal 14.

a) Δlat_i is computed from the equation:

$$\Delta lat_i = \frac{360°}{4 \cdot NZ - i}$$

b) The latitude zone index number, j, is then computed from the values of lat_s, Δlat_i and YZ_i using the equation:

$$j = \text{floor}\left(\frac{lat_s}{\Delta lat_i}\right) + \text{floor}\left(\frac{1}{2} + \frac{\text{MOD}(lat_s, \Delta lat_i)}{\Delta lat_i} - \frac{YZ_i}{2^{Nb}}\right)$$

c) The decoded position latitude, $Rlat_i$, is then computed from the values of j, Δlat_i, and YZ_i using the equation:

$$Rlat_i = \Delta lat_i \cdot \left(j + \frac{YZ_i}{2^{Nb}}\right)$$

d) Δlon_i (the longitude zone size in the E-W direction) is then computed from $Rlat_i$ using the equation:

$$\Delta lon_i = \begin{cases} \dfrac{360°}{\text{NL}(Rlat_i) - i}, & \text{when } \text{NL}(Rlat_i) - i > 0 \\ 360°, & \text{when } \text{NL}(Rlat_i) - i = 0 \end{cases}$$

e) The longitude zone coordinate m is then computed from the values of lon_s, Δlon_i, and XZ_i using the equation:

$$m = \text{floor}\left(\frac{lon_s}{\Delta lon_i}\right) + \text{floor}\left(\frac{1}{2} + \frac{\text{MOD}(lon_s, \Delta lon_i)}{\Delta lon_i} - \frac{YZ_i}{2^{Nb}}\right)$$

f) The decoded position longitude, $Rlon_i$, is then computed from the values of m, XZ_i, and Δlon_i using the equation:

$$Rlon_i = \Delta lon_i \cdot \left(m + \frac{YZ_i}{2^{Nb}}\right)$$

2.6.6 Computations for the surface message

The following computations shall be performed to obtain the decoded latitude and longitude for the surface position format.

a) Δlat_i is computed from the equation:

$$\Delta lat_i = \frac{90°}{4 \cdot NZ - i}$$

b) The latitude zone index, j, is then computed from the values of lat_s, Δlat_i and YZ_i using the equation:

$$j = \text{floor}\left(\frac{lat_s}{\Delta lat_i}\right) + \text{floor}\left(\frac{1}{2} + \frac{\text{MOD}(lat_s, \Delta lat_i)}{\Delta lat_i} - \frac{YZ_i}{2^{17}}\right)$$

c) The decoded position latitude, $Rlat_i$, is then computed from the values of j, Δlat_i, and YZ_i using the equation:

$$Rlat_i = \Delta lat_i \cdot \left(j + \frac{YZ_i}{2^{17}}\right)$$

d) Δlon_i (the longitude zone size, in the E-W direction) is then computed from $Rlat_i$ using the equation:

$$\Delta lon_i = \begin{cases} \dfrac{90°}{\text{NL}(Rlat_i) - i}, & \text{when } \text{NL}(Rlat_i) - i > 0 \\ 90°, & \text{when } \text{NL}(Rlat_i) - i = 0 \end{cases}$$

e) The longitude zone coordinate m is then computed from the values of lon_s, Δlon_i, and XZ_i using the equation:

$$m = \text{floor}\left(\frac{lon_s}{\Delta lon_i}\right) + \text{floor}\left(\frac{1}{2} + \frac{\text{MOD}(lon_s, \Delta lon_i)}{\Delta lon_i} - \frac{YZ_i}{2^{17}}\right)$$

f) The decoded position longitude, $Rlon_i$, is then computed from the values of m, XZ_i, and Δlon_i using the equation:

$$Rlon_i = \Delta lon_i \cdot \left(m + \frac{YZ_i}{2^{17}}\right)$$

2.6.7 GLOBALLY UNAMBIGUOUS AIRBORNE POSITION DECODING

The CPR algorithm shall utilize one airborne-encoded "**even**" format reception (denoted XZ_0, YZ_0), together with one airborne-encoded "**odd**" format reception (denoted XZ_1, YZ_1), to regenerate the global geographic position latitude, *Rlat*, and longitude, *Rlon*. The time between the "**even**" and "**odd**" format encoded position reports shall be no longer than 10 seconds.

Note 1.— This algorithm might be used to obtain globally unambiguous position reports for aircraft out of the range of ground sensors, whose position reports are coming via satellite data links. It might also be applied to ensure that local positions are being correctly decoded over long ranges from the receiving sensor.

Note 2.— The time difference limit of 10 seconds between the even- and odd-format position reports is determined by the maximum permitted separation of 3 NM. Positions greater than 3 NM apart cannot be used to solve a unique global position. An aircraft capable of a speed of 1 850 km/h (1 000 kt) will fly about 5.1 km (2.8 NM) in 10 seconds. Therefore, the CPR algorithm will be able to unambiguously decode its position over a 10-second delay between position reports.

Given a 17-bit airborne position encoded in the "**even**" format (XZ_0, YZ_0) and another encoded in the "**odd**" format (XZ_1, YZ_1), separated by no more than 10 seconds (= 3 NM), the CPR algorithm shall regenerate the geographic position from the encoded position reports by performing the following sequence of steps:

a) Compute Δlat_0 and Δlat_1 from the equation:

$$\Delta lat_i = \frac{360°}{4 \cdot NZ - i}$$

b) Compute the latitude index:

$$j = \text{floor}\left(\frac{59 \cdot YZ_0 - 60 \cdot YZ_1}{2^{17}} + \frac{1}{2}\right)$$

c) Compute the values of $Rlat_0$ and $Rlat_1$ using the following equation:

$$Rlat_i = \Delta lat_i \cdot \left(\text{MOD}(j{,}60 - i) + \frac{YZ_i}{2^{17}}\right)$$

Southern hemisphere values of $Rlat_i$ will fall in the range from 270° to 360°. Subtract 360° from such values, thereby restoring $Rlat_i$ to the range from –90° to +90°.

d) If **NL**($Rlat_0$) is not equal to **NL**($Rlat_1$) then the two positions straddle a transition latitude, thus a solution for global longitude is not possible. Wait for positions where they are equal.

e) If **NL**($Rlat_0$) is equal to **NL**($Rlat_1$) then proceed with computation of Δlon_i, according to whether the most recently received airborne position message was encoded with the even format ($i = 0$) or the odd format ($i = 1$) :

$$\Delta lon_i = \frac{360°}{n_i},$$

where n_i = greater of [NL($Rlat_i$) – i] and 1.

f) Compute m, the longitude index:

$$m = \text{floor}\left(\frac{XZ_0 \cdot (NL - 1) - XZ_1 \cdot NL}{2^{17}} + \frac{1}{2}\right),$$

where NL = NL ($Rlat_i$).

g) Compute the global longitude, $Rlon_0$ or $Rlon_1$, according to whether the most recently received airborne position message was encoded using the even format (that is, with $i = 0$) or the odd format ($i = 1$):

$$Rlon_i = \Delta lon_i \cdot \left(\text{MOD}(m{,}n_i) + \frac{YZ_i}{2^{17}}\right),$$

where n_i = greater of [NL($Rlat_i$) – i] and 1.

2.6.8 CPR DECODING OF RECEIVED POSITION REPORTS

2.6.8.1 *OVERVIEW*

The techniques described in the preceding paragraphs (locally and globally unambiguous decoding) shall be used together to decode the latitude/longitude contained in airborne, surface, and TCP or TCP+1 position reports. The process shall begin with globally unambiguous decoding based upon the receipt of an even and an odd encoded position squitter. Once the globally unambiguous position is determined, either of two approaches shall be used to support subsequent decoding based upon a single position report, either even or odd encoding. The two techniques shall be range monitoring and emitter centered local decoding.

2.6.8.2 *RANGE MONITORING LOCAL DECODING*

2.6.8.2.1 *Range monitoring technique*

In this approach, local decoding for the airborne format (2.6.4) shall be performed based upon the current position of the receiver. This shall provide the position of a transmitting aircraft that is unambiguous to plus or minus 180 NM.

Note 1.— If the transmitting aircraft is within 180 NM, the local decoding technique will correctly decode the location of the aircraft.

The range of the transmitting aircraft shall be checked at detection and tracks shall only be initiated if the range is less than 180 NM. Once initiated, the range of the tracked aircraft shall be checked at each update and the track shall be dropped if the range becomes equal to or greater than 180 NM.

For the surface format, the same process shall be used except that the transmitting aircraft must be within 45 NM for detection and tracking.

Note 2.— The range limits are reduced since the ambiguity limit for the surface position reports is one-fourth that of the airborne case.

2.6.8.2.2 *Range monitoring example*

2.6.8.2.2.1 Decoding of airborne position

2.6.8.2.2.1.1 *Detection.* At detection, a globally unambiguous decode shall be performed. If range is greater than 160 NM, the detection attempt shall be discontinued and the track information discarded. Detection shall be attempted if squitters continue to be received. If the globally decoded range remains greater than 160 NM, the track information shall continue to be discarded.

Note.— If the aircraft is approaching, detection will succeed when the range decreases to less than or equal to 160 NM.

2.6.8.2.2.1.2 *Track monitoring.* After detection, range shall be monitored during each surveillance update. If range is greater than 170 NM, the track shall be dropped.

Note.— The use of 160 NM for detection and 170 NM for track drop provides hysteresis that avoids reacquiring a track that was just dropped due to long range. Thus a track dropped at 170 NM would not be reacquired unless its range dropped to less than or equal to 160 NM.

2.6.8.2.2.2 *Decoding of surface position.* Using the range monitoring technique for decoding squitters in the surface format, the same process as above shall be used except that the track shall be initiated at 40 NM and dropped at 42.5 NM.

2.6.8.2.3 Emitter centered local decoding

In this approach, the most recent position of the emitter shall be used as the basis for the local decoding.

Note.— This produces an unambiguous decoding at each update, since the transmitting aircraft cannot move more than 360 NM between position updates.

2.6.8.2.4 *Technique application*

The range monitoring technique shall only be used for ranges less than or equal to 180 NM, for example in air-to-air applications. For ground stations (i.e. non-aircraft implementations) that are required to operate at ranges in excess of 180 NM, only the emitter centred technique can shall be used.

Note.— The emitter centered technique can be used for both airborne receivers and ground stations.